

Final Mobile Application Development Group Project Report

Project Title:

Android Healthcare Project

Team Members:

1. Hriday Shah (Registration, Find doctor, health articles, logout)
2. Vidhie Jhunjhunwala (Login, Lab Test, buy medicine, order details)

1. Introduction:

This project ,**healthcare application** , aims to provide essential healthcare services on demand, available 24/7. The purpose of the app is to offer a one-stop solution for users seeking convenient access to various medical services. The problem it addresses is the lack of readily accessible, comprehensive healthcare information and services, especially for those who may have limited access to in-person healthcare resources.

Key Features:

1. **Book Appointment** - Allows users to schedule appointments with healthcare providers.
2. **Buy Medicine** - Provides a way for users to purchase medicines online.
3. **Find Doctor** - Enables users to search for doctors based on their needs.
4. **Lab Tests** - Offers lab test bookings and details about tests available.

5. **Health Articles** - Contains informative articles to educate users on various health topics.
6. **Order Details** - Allows users to view their medicine and test order history.

Target Users:

The target users are likely individuals who want to manage their healthcare needs from the convenience of their homes. This includes people with chronic health conditions, busy professionals, and anyone looking to streamline their healthcare management.

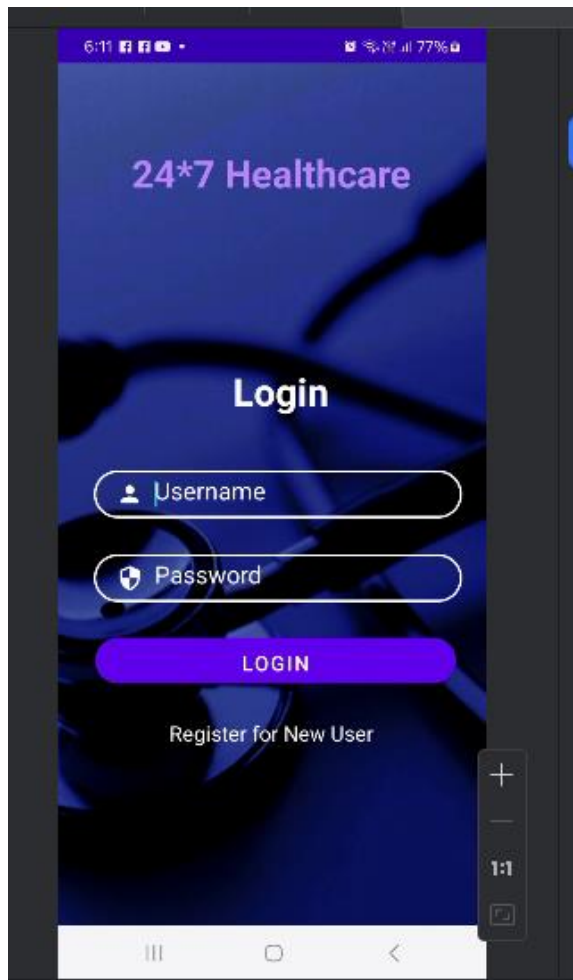
Vision:

The vision of the application is to provide an all-encompassing platform where users can manage appointments, purchase medications, and stay informed about health-related topics, contributing to better health outcomes and convenience.

2. Design and UI (15%)

2.1 User Interface:

Login Page:



6:11 77%

24*7 Healthcare

Login

Username

Password

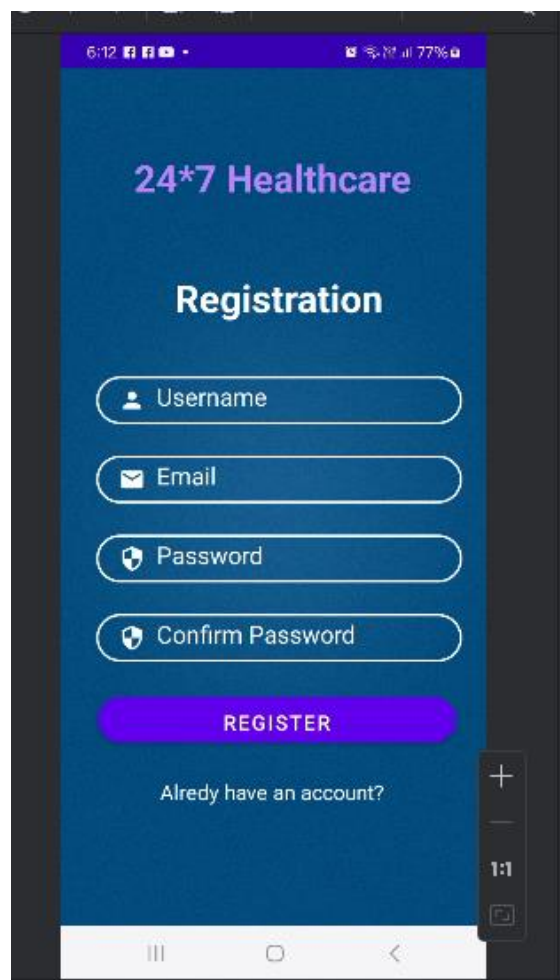
LOGIN

Register for New User

1:1

The login page features a dark blue background with a medical-themed image. It includes a status bar at the top showing the time as 6:11 and battery level at 77%. The page title '24*7 Healthcare' is at the top, followed by the 'Login' heading. There are two input fields for 'Username' and 'Password', each with a corresponding icon. A purple 'LOGIN' button is positioned below the fields. A link 'Register for New User' is at the bottom. A vertical toolbar on the right contains a plus sign, a minus sign, a '1:1' zoom indicator, and a close button. The bottom of the screen shows standard Android navigation icons.

Registration Page:



6:12 77%

24*7 Healthcare

Registration

Username

Email

Password

Confirm Password

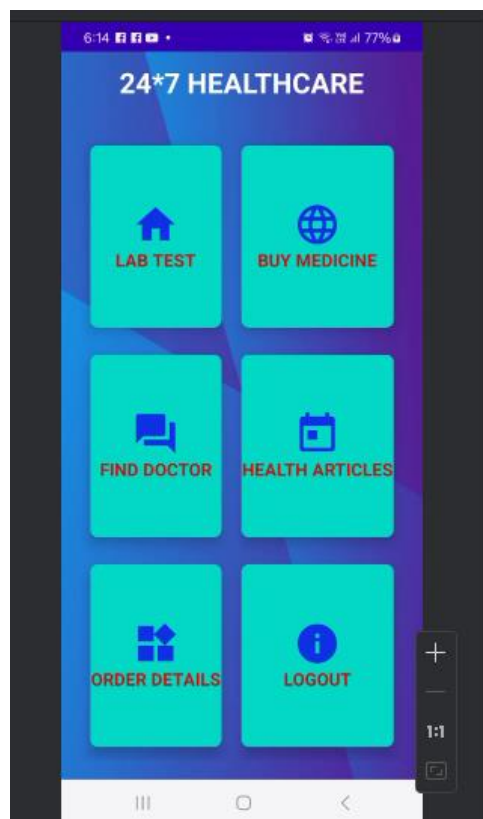
REGISTER

Alredy have an account?

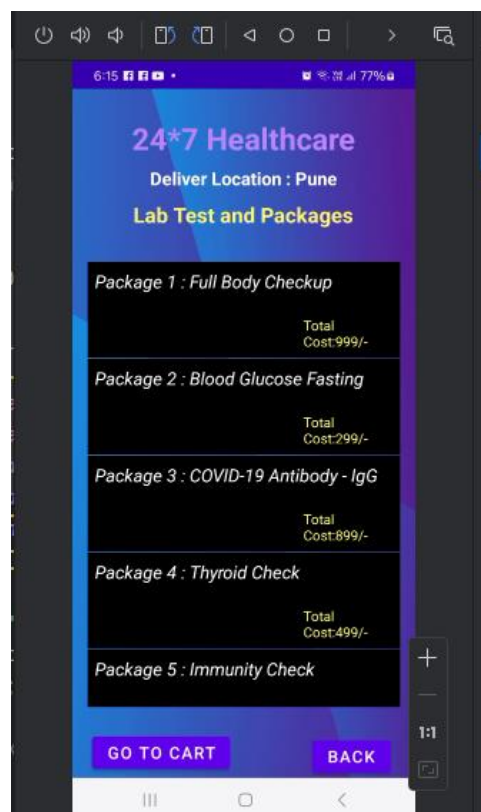
1:1

The registration page has a dark blue background with a medical-themed image. It includes a status bar at the top showing the time as 6:12 and battery level at 77%. The page title '24*7 Healthcare' is at the top, followed by the 'Registration' heading. There are four input fields for 'Username', 'Email', 'Password', and 'Confirm Password', each with a corresponding icon. A purple 'REGISTER' button is positioned below the fields. A link 'Alredy have an account?' is at the bottom. A vertical toolbar on the right contains a plus sign, a minus sign, a '1:1' zoom indicator, and a close button. The bottom of the screen shows standard Android navigation icons.

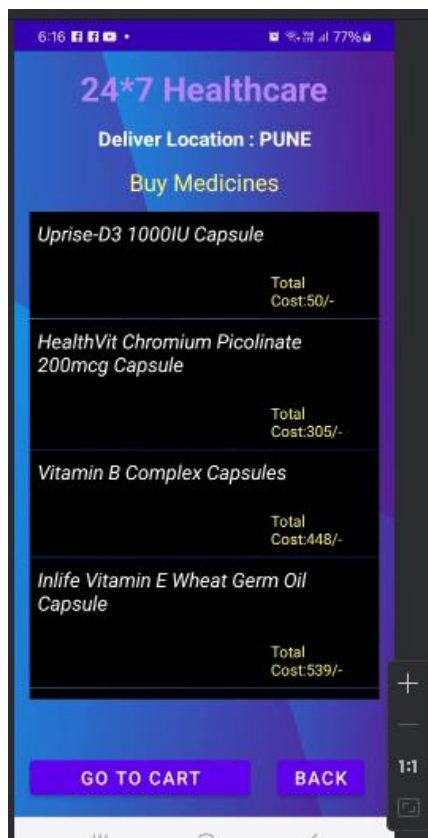
HomePage:



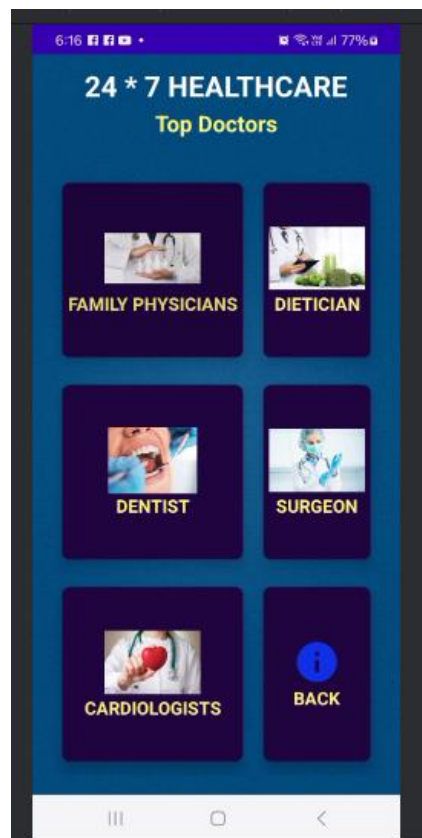
LabTest Page:



BuyMedicine Page:

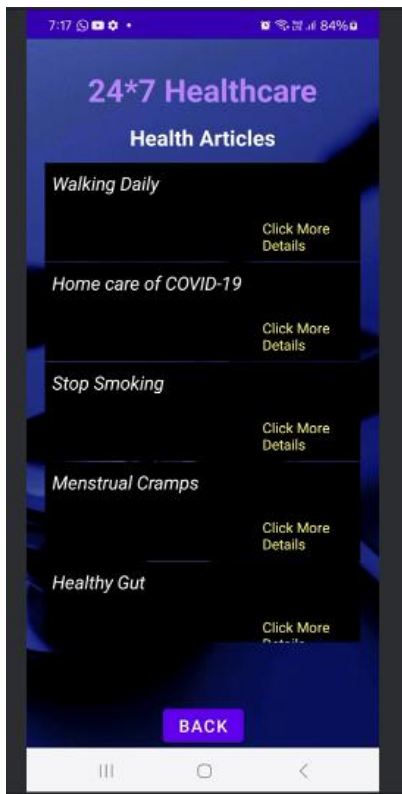


FindDoctor Page:



Health Article Page:

Order Details Page:



Here are some of the key screens in the application:

1. **Home Screen:** The home screen (as shown in the screenshot provided) displays the main services offered, such as **Lab Test**, **Buy Medicine**, **Find Doctor**, **Health Articles**, **Order Details**, and **Logout**. Each option is represented as a clickable card with an icon and label.
2. **Login Screen:** A login screen where users can authenticate themselves by entering credentials. This screen ensures that the user's information remains private and secure.
3. **Dashboard:** After login, users are directed to the main dashboard (or Home screen), where they can access all essential services directly.
4. **Service Details Screens:** Separate screens for each service, such as booking lab tests, ordering medicines, finding doctors, etc. Each screen provides a straightforward interface for users to complete their tasks without unnecessary complexity.

Design Considerations:

- **Color Scheme:** The color scheme uses a vibrant blue and teal combination with white text, aiming for a fresh and modern look that aligns with healthcare themes. These colors also provide high contrast, which improves readability.

- **Fonts:** Clean, sans-serif fonts are used to ensure readability and a professional appearance. The font size is optimized for easy reading, especially on smaller screens.
- **Icons and Layouts:** Each service is represented by intuitive icons (e.g., a house icon for "Home," a medicine bottle for "Buy Medicine") to aid in quick recognition. The layout is grid-based, creating an organized and easy-to-navigate interface.

2.2 User Flow:

- A diagram showing the flow between different screens and how users interact with the app.

Here's an overview of the user flow:

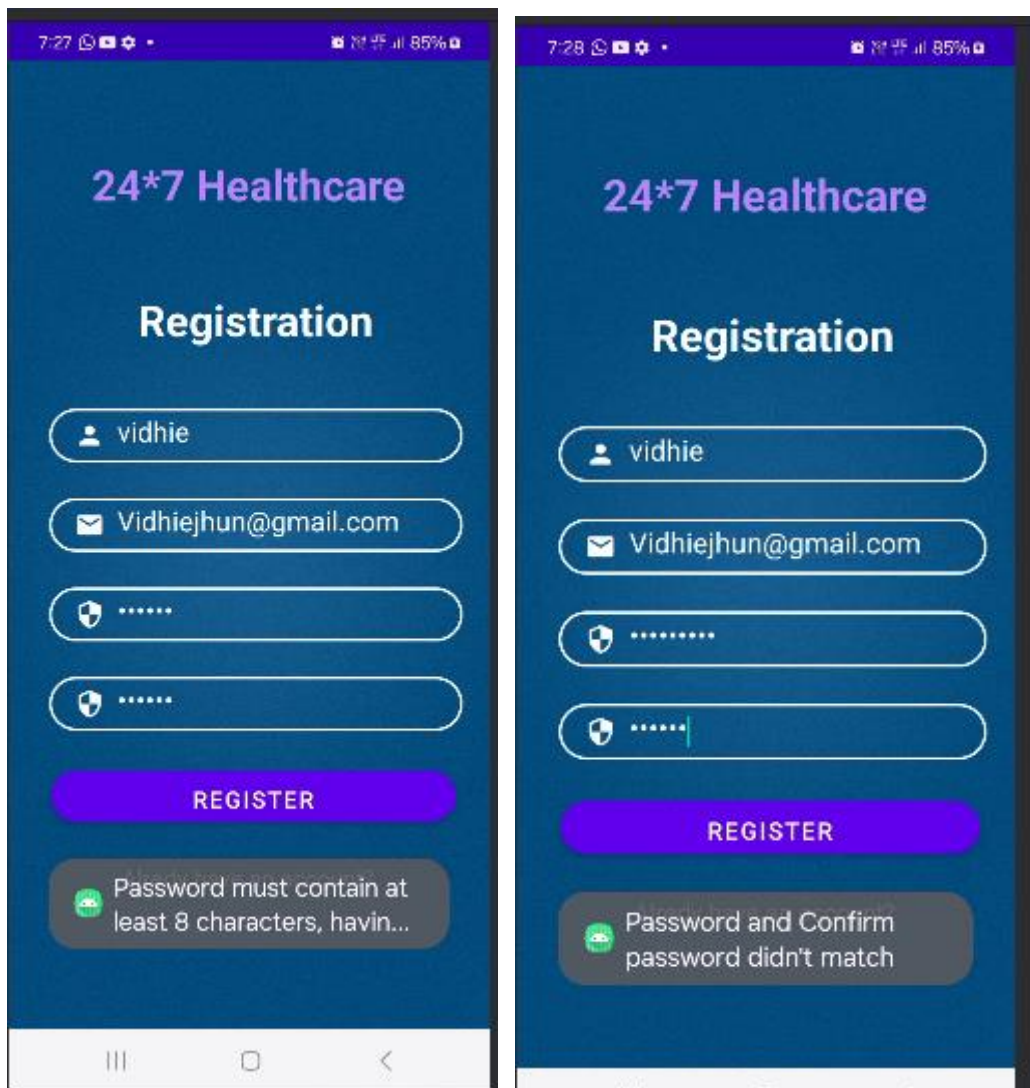
1. **Login/Register** -> Users are authenticated upon entering credentials.
2. **Home/Dashboard Screen** -> After login, users access the main dashboard where they can see all services.
3. **Navigation:**
 - **Book Appointment** -> Selects a doctor and sets an appointment time.
 - **Buy Medicine** -> Navigates to a list of available medicines and makes a purchase.
 - **Find Doctor** -> Searches for doctors by specialty or location.
 - **Lab Tests** -> Views available lab tests and books an appointment.
 - **Health Articles** -> Reads articles on health and wellness topics.
 - **Order Details** -> Views order history for lab tests or medicine.
4. **Logout** -> Logs out, returning the user to the login screen

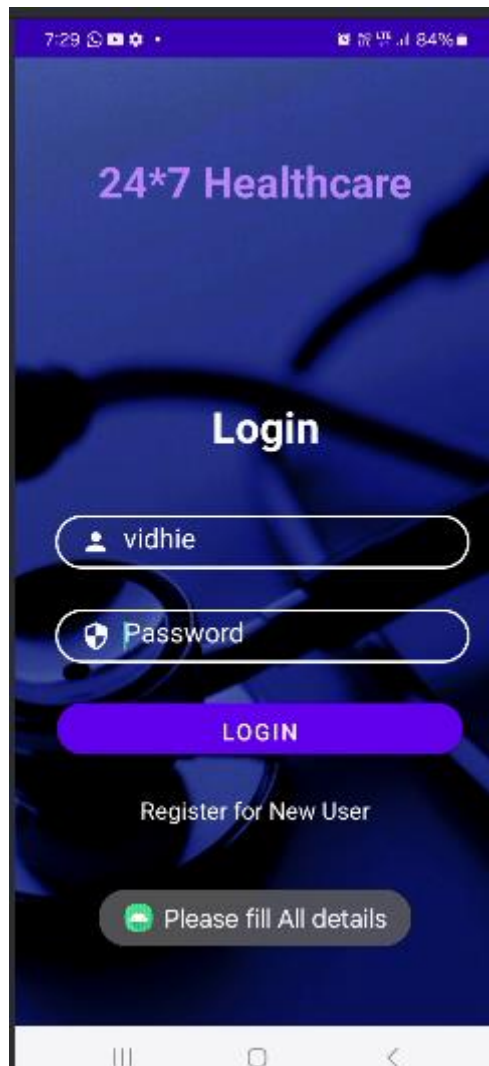
3. Functionality (20%)

3.1 Core Features:

1. User Authentication and Registration

- **Description:** Allows new users to register by providing personal details, and existing users to log in securely. The authentication process could involve verifying email addresses or phone numbers.
- **How It Works:**
 - Registration requires the user to enter their details (e.g., name, email, password).
 - Login verifies the credentials and directs the user to the home screen upon successful authentication.



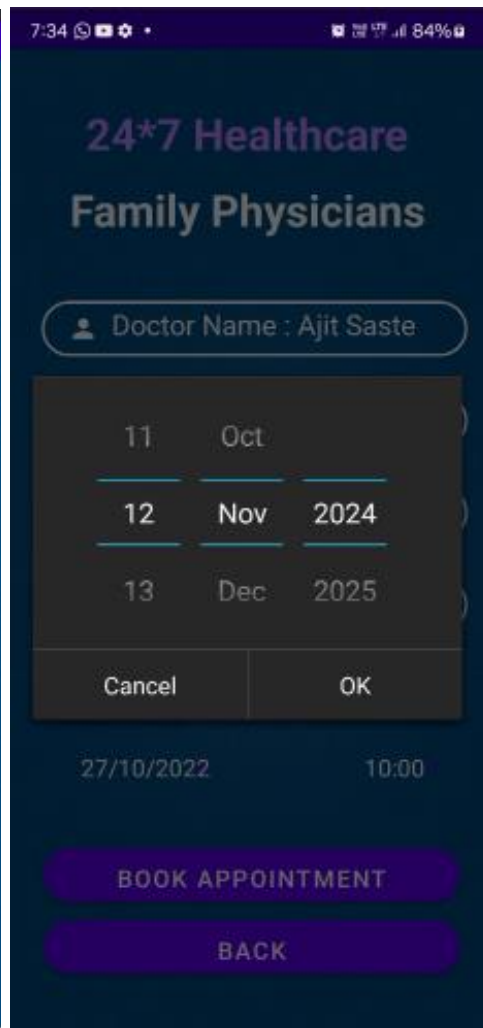




2. Doctor Search and Appointment Booking

- **Description:** Enables users to find doctors by specialty and book appointments directly through the app.
- **How It Works:**
 - The app provides a categorized list of doctors by specialties, such as Family Physicians, Dieticians, Dentists, Surgeons, and Cardiologists.
 - Users can select a specialty, which will lead to a list of available doctors within that category, showing each doctor's profile.
 - After selecting a doctor, users can book an appointment by choosing an available time slot.







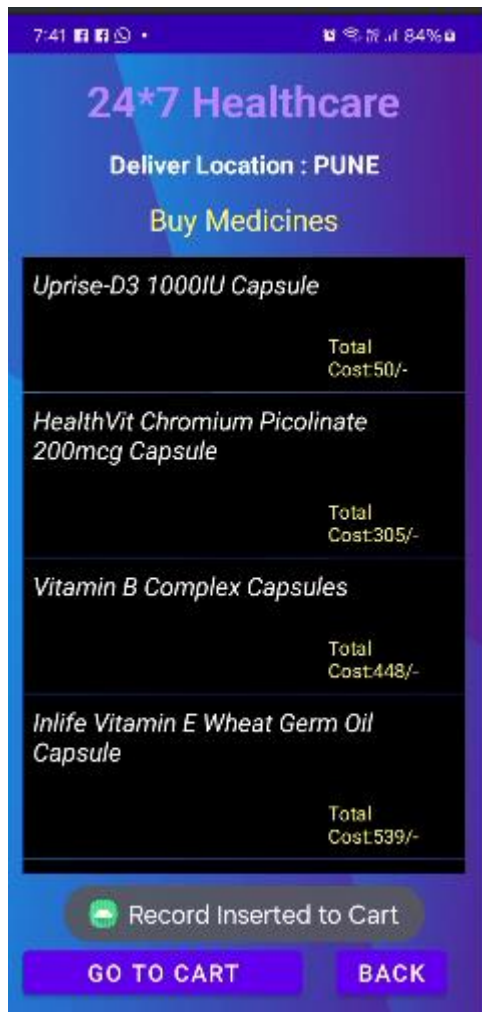
3. Medicine Ordering System

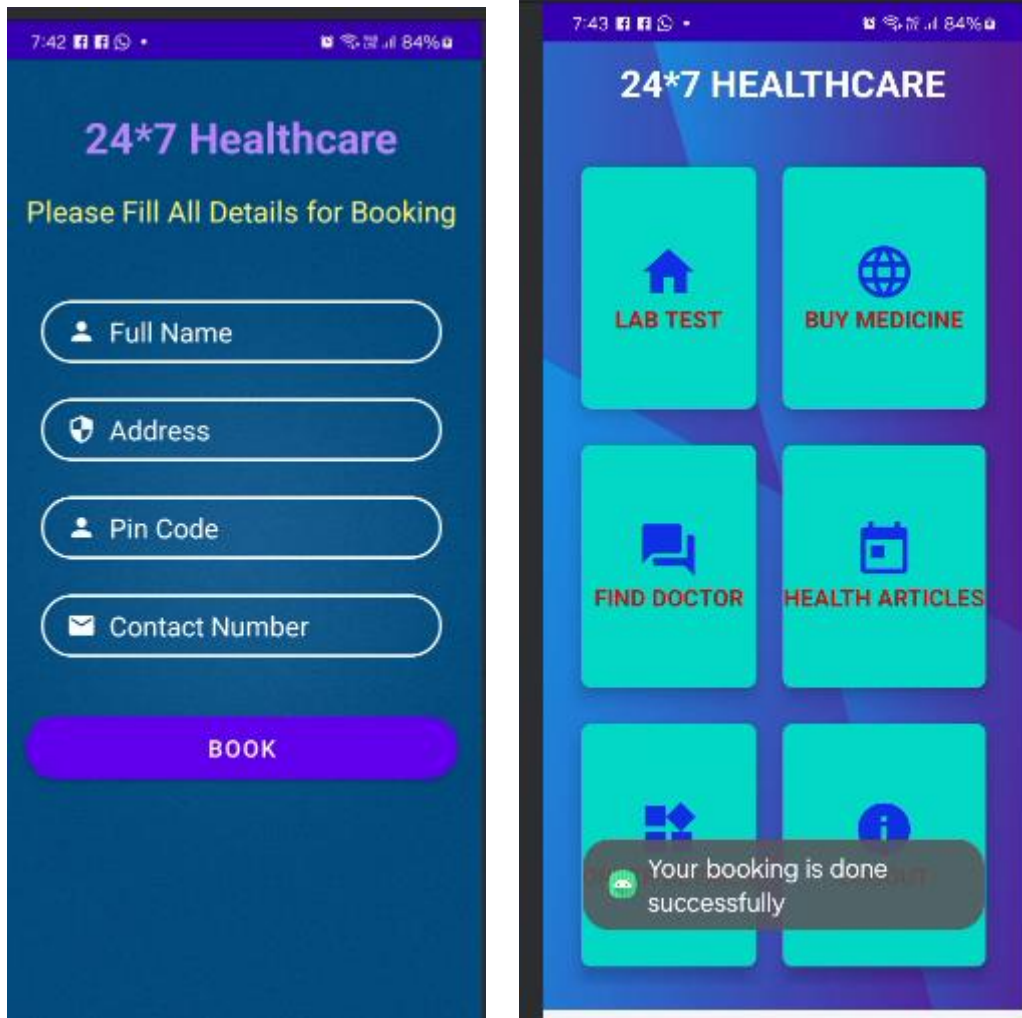
Description: This feature enables users to order medicines online and have them delivered to their specified location.

How It Works:

- Users can browse or search for specific medicines.
- Each medicine displays details, including name and total cost.
- Users can add selected items to a cart.
- Orders can be placed for delivery to the chosen location (in this example, "PUNE").

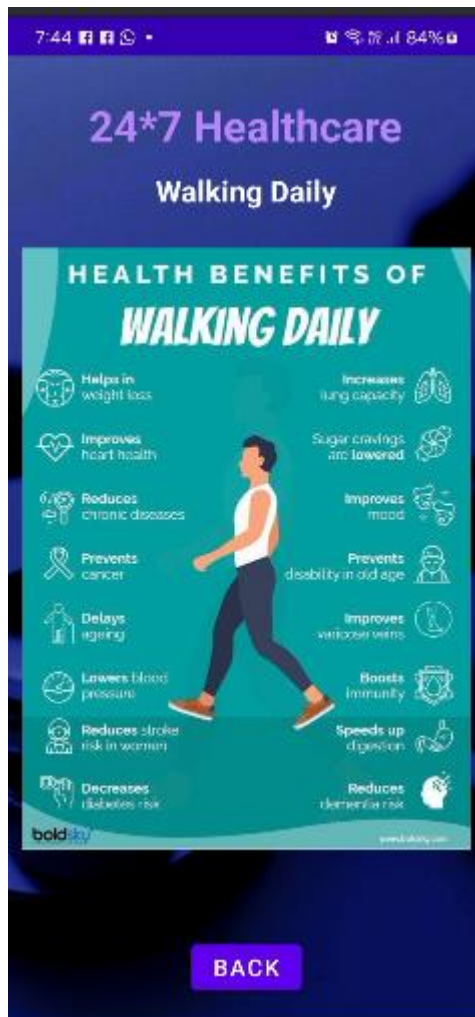
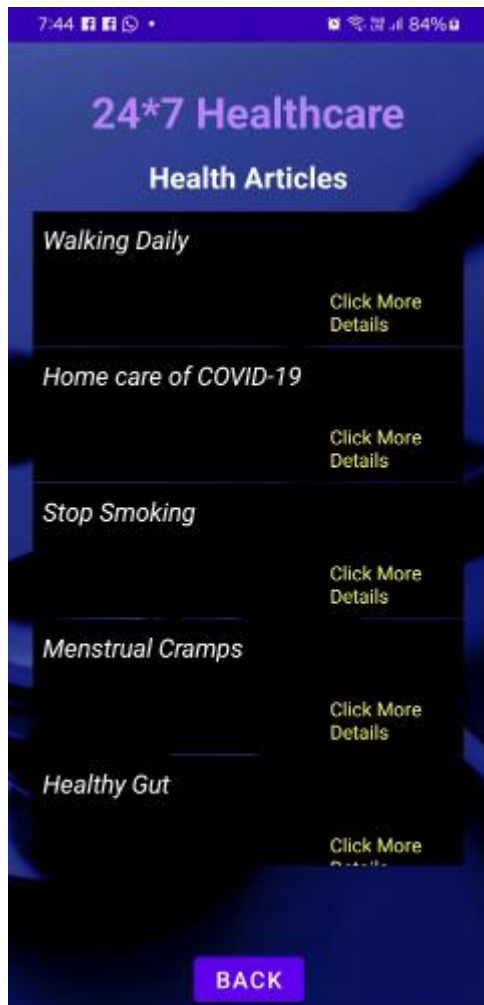






4. Health Articles and Resources

- **Description:** Provides users with access to a library of health-related articles, news, and tips to promote wellness and health education.
- **How It Works:**
 - Articles are categorized by topics (e.g., nutrition, mental health, fitness).
 - Users can browse, read, and share articles, keeping them informed about health trends.



5. Lab Test and Packages

Description: This feature allows users to book various health check-up packages online and have tests scheduled at their convenience.

How It Works:

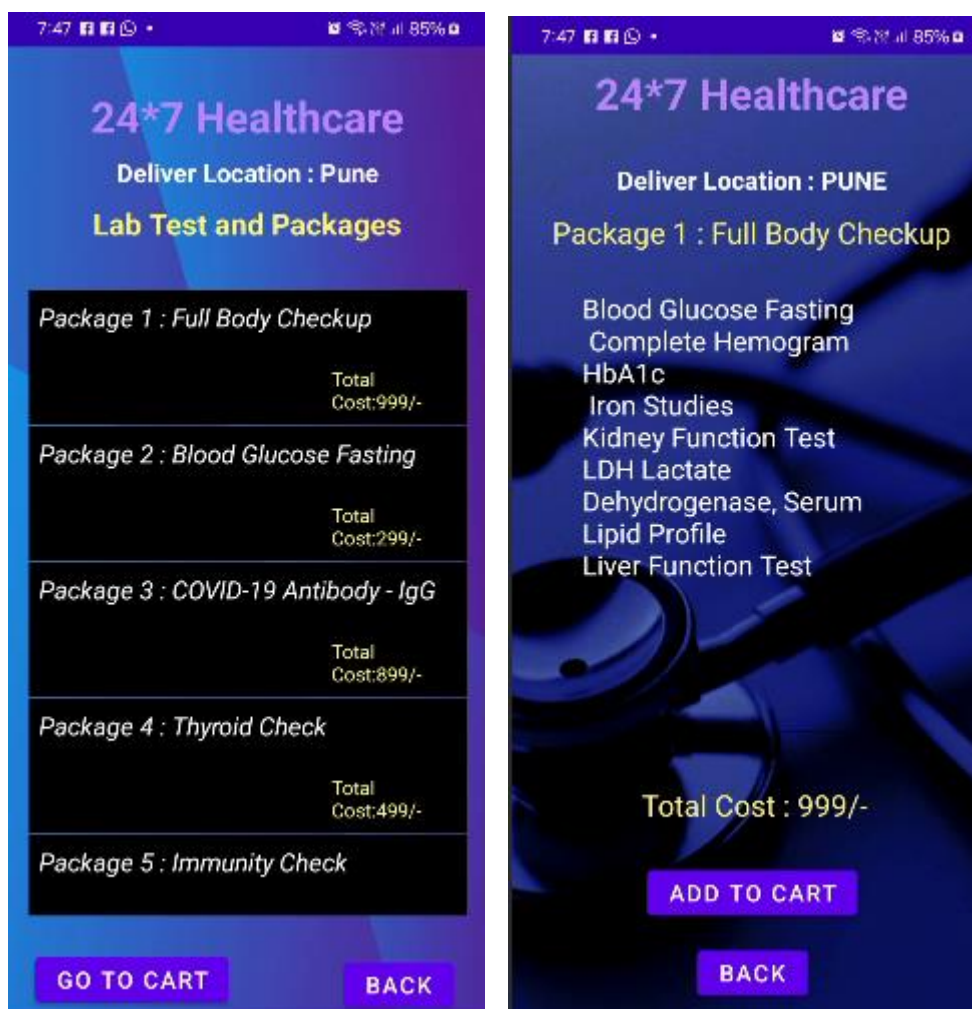
- Users can browse through different lab test packages available for booking.
- Each package shows the test name and total cost.
- Users can add selected packages to a cart and proceed to schedule.

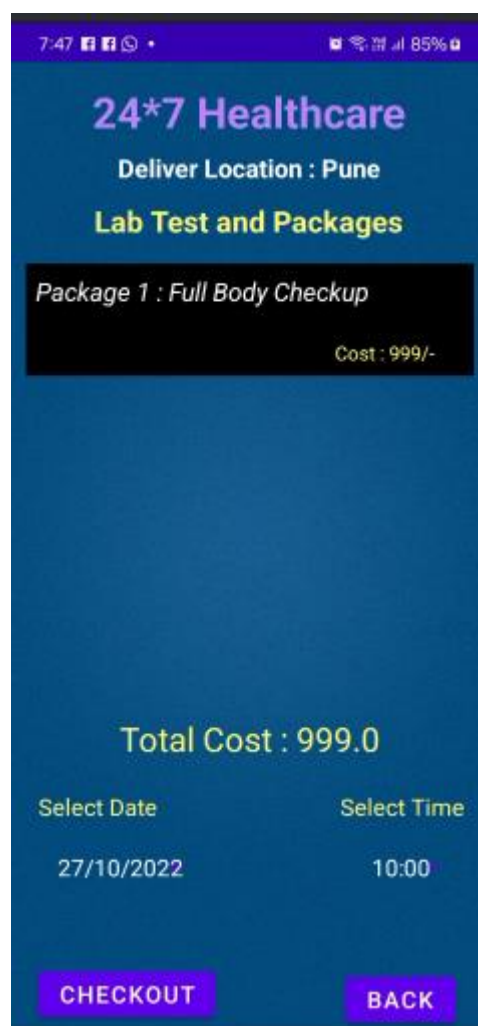
- The app integrates with healthcare providers or labs to ensure real-time availability and scheduling options.

Example Packages:

- Package 1: Full Body Checkup - ₹999
- Package 2: Blood Glucose Fasting - ₹299
- Package 3: COVID-19 Antibody - IgG - ₹899
- Package 4: Thyroid Check - ₹499
- Package 5: Immunity Check - ₹699

Options: Users can proceed to the cart to review their selections or go back to browse more options







6. Order Details

Description: This feature provides users with a summary of their healthcare orders, including appointments, medicine orders, and lab tests, along with relevant details such as doctor names, delivery dates, and locations.

How It Works:

- Users can view the details of each order type (appointments, medicines, lab tests) in one place.
- Each entry includes specifics like the doctor's name, hospital address, delivery date, and time for appointments, as well as the order type and cost for medicines and lab tests.

7:51

85%

24*7 Healthcare

ORDER DETAILS

Family Physicians => Doctor Name :
Ajit Saste

Hospital Address : Pimpri

Rs.600 Del:12/11/2024 10:00 appointment

vidhie

xyz

Rs.355 Del:27/10/2022 medicine

vidhie

xyz

Rs.999 Del:27/10/2022 10:00 lab

BACK

7. Logout and Security Settings

Description: Allows users to log out and manage security settings like password reset and account management.

- **How It Works:**
 - Users can log out from their account and manage security settings from the settings page



4. Innovation & Creativity (10%)

4.1 Unique Features or Approaches:

1. Specialty-Based Doctor Navigation

- **Innovative Aspect:** Instead of requiring users to type in search terms, the app offers a visually guided, specialty-based doctor selection, as seen in the "Top Doctors" screen. Each medical specialty is represented by an image and label, allowing users to quickly identify and select the type of healthcare professional they need.
- **Why It Stands Out:** This approach simplifies the user experience, particularly for users who may be unfamiliar with medical terminology. By using visual cues, it makes the app more accessible and user-friendly, especially for older adults or those less comfortable with typing.

2. 24/7 Access and Instant Booking

- **Innovative Aspect:** The app emphasizes "24 * 7 Healthcare," allowing users to find and book appointments at any time. Appointments can be made with just a few taps, giving users flexibility and immediacy in their healthcare access.
- **Why It Stands Out:** This real-time booking feature stands out by prioritizing user convenience and ensuring healthcare support is available around the clock. It reflects a commitment to continuous access, which is particularly valuable in urgent situations.

3. Integrated Medicine Ordering and Lab Test Booking

- **Innovative Aspect:** The app integrates multiple healthcare needs in one place—users can not only book appointments but also order medicines and schedule lab tests. This all-in-one approach saves users the hassle of navigating multiple platforms for different healthcare needs.
- **Why It Stands Out:** The comprehensive approach enhances convenience and promotes continuity of care, making it easier for users to manage their healthcare activities from a single app. This holistic design is particularly beneficial for patients with chronic conditions who need regular medication and lab tests.

4. Health Education Resources and Articles

- **Innovative Aspect:** The app offers a library of health-related articles and resources, allowing users to stay informed about various health topics and trends. Articles are categorized by themes (e.g., fitness, nutrition, mental health) for easy navigation.
 - **Why It Stands Out:** By combining health information with service access, the app promotes not only immediate healthcare access but also long-term health literacy and preventative care. This educational component encourages proactive health management.
-

5. Technical Complexity (15%)

5.1 Technical Challenges:

Creating and managing an efficient and secure database for a healthcare application involves several technical challenges. Here are some of the most challenging parts of implementing the SQLite database in this app, including code snippets that demonstrate these complex functionalities.

1. User Authentication and Data Security

- **Challenge:** Ensuring secure storage of user information, especially sensitive data like usernames, emails, and passwords, requires careful handling to prevent unauthorized access and data breaches.
- **Solution:** The register and login functions allow for basic user authentication

CodeSnippet:

```
public void register(String username, String email, String password) {
    ContentValues cv = new ContentValues();
    cv.put("username", username);
    cv.put("email", email);
    cv.put("password", password); // In a real app, hash the password
    SQLiteDatabase db = getWritableDatabase();
    db.insert("users", null, cv);
    db.close();
}

public int login(String username, String password) {
    int result = 0;
    String[] str = {username, password};
    SQLiteDatabase db = getReadableDatabase();
    Cursor c = db.rawQuery("select * from users where username=? and password=?", str);
    if (c.moveToFirst()) {
        result = 1; // User found
    }
    c.close();
    db.close();
    return result;
}
```

- **Explanation:** This code registers users and allows login by verifying credentials. For enhanced security, it's recommended to hash passwords before storage.

2. Cart Management System

- **Challenge:** Managing a cart that allows users to add or remove items, ensuring that each user's cart data is stored independently, and providing real-time updates without duplicating items.
- **Solution:** The app implements methods to add items to the cart, check for existing items, remove items, and retrieve cart data for each user.

CodeSnippet:

```

public void addCart(String username, String product, float price, String otype) {
    ContentValues cv = new ContentValues();
    cv.put("username", username);
    cv.put("product", product);
    cv.put("price", price);
    cv.put("otype", otype);
    SQLiteDatabase db = getWritableDatabase();
    db.insert("cart", null, cv);
    db.close();
}

public int checkCart(String username, String product) {
    int result = 0;
    String[] str = {username, product};
    SQLiteDatabase db = getReadableDatabase();
    Cursor c = db.rawQuery("select * from cart where username = ? and product = ?", str);
    if (c.moveToFirst()) {
        result = 1; // Item already in cart
    }
    c.close();
    db.close();
    return result;
}

```

3. Order Placement and Appointment Scheduling

- **Challenge:** Implementing a feature to schedule appointments and manage orders, while checking if an appointment already exists to avoid double-booking.
- **Solution:** The app uses the addOrder function to place an order and the checkAppointmentExists function to verify if a similar appointment already exists.

```

public void addOrder(String username, String fullname, String address, String contact, int
    ContentValues cv = new ContentValues();
    cv.put("username", username);
    cv.put("fullname", fullname);
    cv.put("address", address);
    cv.put("contactno", contact);
    cv.put("pincode", pincode);
    cv.put("date", date);
    cv.put("time", time);
    cv.put("amount", price);
    cv.put("otype", otype);
    SQLiteDatabase db = getWritableDatabase();
    db.insert("orderplace", null, cv);
    db.close();
}

public int checkAppointmentExists(String username, String fullname, String address, String
    int result = 0;
    String[] str = {username, fullname, address, contact, date, time};
    SQLiteDatabase db = getReadableDatabase();
    Cursor c = db.rawQuery("select * from orderplace where username = ? and fullname = ? a
    if (c.moveToFirst()) {
        result = 1; // Appointment exists
    }
    c.close();
    db.close();
    return result;
}

```

6. Security and Data Management (10%)

6.1 Data Handling:

In this healthcare application, data management is centered around the use of SQLite, a lightweight relational database embedded within Android. Here's an overview of how data is managed, stored, and secured:

1. SQLite Database:

- **Purpose:** SQLite is used for storing essential data such as user profiles, cart information, and order/appointment details.
- **Tables:** There are three main tables in the database:
 - **Users Table:** Stores user credentials (username, email, password).
 - **Cart Table:** Holds information about items added to the cart by users, including product names, prices, and order types.
 - **OrderPlace Table:** Manages data related to placed orders or scheduled appointments, such as user contact details, addresses, dates, and times.

2. Data Storage and Management:

- **User Authentication:** User data, including usernames, emails, and passwords, is stored in the users table to enable login and authentication.
- **Cart Management:** The cart table is updated whenever a user adds or removes items from their cart.
- **Order/Appointment Management:** Orders and appointments are stored in the orderplace table. This table is checked for existing appointments to avoid double-booking.

3. Data Format and Access:

- **Local Data Storage:** SQLite stores data locally on the device, allowing offline access and faster read/write operations.
- **Data Access Methods:** The app uses SQL queries to interact with the database, including insertion, deletion, and retrieval of data. These queries are encapsulated within methods, ensuring data is accessed in a structured manner.

6.2 Security Measures:

To protect user data, several security measures are implemented within the app:

1. Password Storage:

- **Hashing (Recommended):** While the current implementation stores plain-text passwords, in a production environment, password hashing (using bcrypt or SHA-256) should be applied to secure passwords. This ensures that even if the database is compromised, user passwords remain protected.

2. Input Validation:

- **User Inputs:** Inputs for registration, login, and order information undergo validation to ensure they meet expected formats and avoid SQL injection attacks. By using parameterized queries (e.g., ? placeholders in SQL statements), the app reduces the risk of SQL injection.

3. Secure Database Access:

- **Restricted Access:** The SQLite database is only accessible within the app's local environment. The database is not exposed externally, reducing the risk of unauthorized access.

4. Authentication and Authorization:

- **Login Verification:** The login method checks the credentials against the database, granting access only to verified users.

7. Testing and Debugging (10%)

7.1 Testing Strategy:

The testing strategy aims to ensure that the healthcare application functions as expected and meets user requirements for accuracy, reliability, and usability. Testing is conducted in several phases to validate different aspects of the application, including functionality, usability, security, and performance.

1. **Unit Testing:** Individual components and modules are tested to ensure each functions correctly in isolation. This step verifies that smaller units of the code are working as intended before integration.
2. **Integration Testing:** After unit testing, different modules are combined and tested together to check if they interact smoothly and deliver the expected outcomes. This step helps detect any interface issues between modules.
3. **System Testing:** The entire application is tested as a whole to evaluate compliance with the specified requirements. This phase involves end-to-end testing to verify that the complete flow, from ordering to delivery, works seamlessly.

7.2 Debugging Process:

➤ Database Connection Issues:

- **Issue:** Crashes when accessing or closing database connections incorrectly.
- **Solution:** Ensured all database connections were closed after each operation, wrapping each in try-catch blocks to handle errors gracefully.
- **Evidence:** Logcat logs confirming successful database operations without crashes.

➤ Login Authentication Failure:

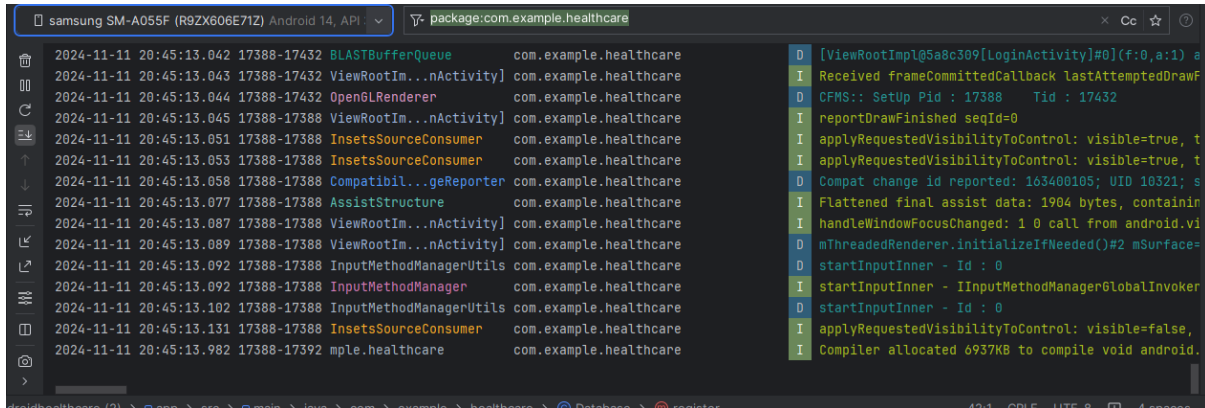
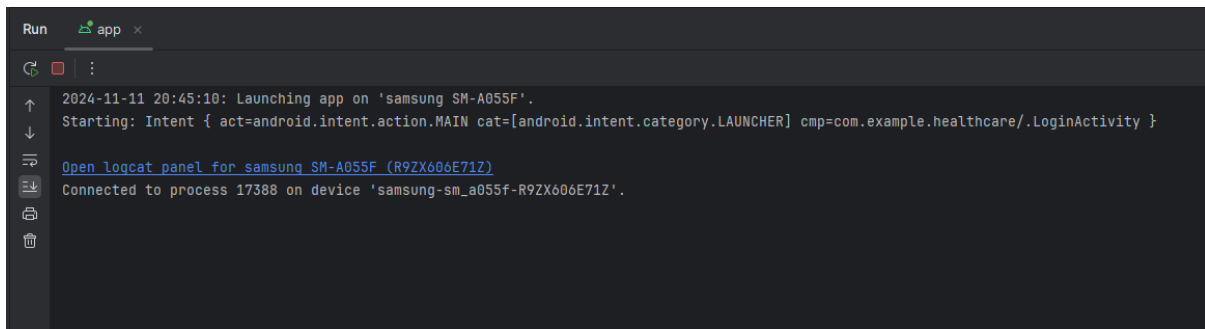
- **Issue:** Incorrect password validation causing login issues for valid users.
- **Solution:** Fixed the SQL query to accurately match usernames and passwords in a case-insensitive manner.
- **Evidence:** Logcat logs showed successful login attempts after applying the fix.

➤ Appointment Duplication:

- **Issue:** Users could create duplicate appointments for the same date and time.
- **Solution:** Added checks in the checkAppointmentExists method to prevent multiple appointments with identical details.
- **Evidence:** Logcat logs confirmed that duplicate appointment entries were blocked, and users received error messages on repeated attempts.

➤ UI Alignment Issues on Different Devices:

- **Issue:** Misalignment of UI components on devices with various screen sizes, causing poor user experience.
- **Solution:** Used **ConstraintLayout** for dynamic resizing and responsive design, ensuring elements aligned correctly on all screen sizes.



8. Team Collaboration (5%)

8.1 Contribution Breakdown:

1. Hriday Shah:

- **Registration Module:** Designed and implemented the user registration functionality, which includes creating a database table for storing user information, validating inputs, and handling user feedback on successful or unsuccessful registrations.
- **Find Doctor Module:** Developed the interface and logic for finding doctors, including searching by specialty, displaying doctor profiles, and appointment booking functionality.
- **Health Articles Module:** Worked on fetching and displaying health-related articles within the app to provide users with helpful health information.
- **Logout Functionality:** Implemented the logout feature, ensuring secure user session management.

- **Testing and Debugging:** Tested modules for bugs, optimized the UI for better usability, and ensured smooth navigation across the app.
- **Documentation:** Contributed to documenting the setup process and detailed descriptions for the modules worked on.

2. Vidhie Jhunhunwala:

- **Login Module:** Developed the login functionality with user authentication, managing database interactions, and handling invalid login attempts.
- **Lab Test Module:** Created the lab test booking functionality, allowing users to select tests, view available slots, and manage their appointments.
- **Buy Medicine Module:** Built the online pharmacy feature where users can browse medicines, add them to their cart, and proceed to checkout.
- **Order Details Module:** Implemented order tracking and history, enabling users to view details of past orders.
- **Testing and Debugging:** Performed integration testing and resolved issues, particularly with user authentication and order management.
- **Documentation:** Documented code and user guides for modules worked on, as well as the testing processes and strategies.

8.2 Tools for Collaboration:

- **GitHub:** Used for version control, managing code commits, branching, and merging contributions from both team members. This ensured code was organized and allowed for easier collaboration.
- **Google Docs:** Used for collaboratively writing and organizing project documentation, including user guides, testing reports, and presentation materials.
- **Android Studio:** Main development environment where both team members worked on their assigned modules. Also used for testing, debugging, and emulating the application on different device sizes.

These collaboration tools helped streamline communication, maintain code integrity, and keep the project on schedule, ensuring a cohesive and fully integrated final product.

9. Documentation (5%)

Submission of this document.

10. Demo & Viva (10%)

Prepare a demonstration video showing your app in action / Give demo on emulator in front of faculty, highlighting its main features, functionality, and how it meets the project requirements. Additionally, be prepared for viva questions regarding your project's technical aspects, teamwork, and decision-making process.

11. Conclusion:

The development process for our healthcare application was divided into several phases, including planning, design, development, testing, and deployment. We began by defining the core features, such as user registration, doctor search, lab test booking, medicine purchases, and health articles. Each team member was assigned specific modules based on their strengths, which helped us work in parallel and maintain efficiency. We employed agile methodologies, regularly meeting to discuss progress, address issues, and update our tasks.

We followed a modular approach to development, ensuring each feature was built independently and integrated smoothly. Android Studio was our main development tool, and we utilized SQLite for data storage and management within the app.

➤ Challenges Faced

During the development, we encountered several technical and logistical challenges:

1. Database Management: Managing data interactions across various modules while ensuring data consistency was challenging. We worked through issues with SQLite, especially when handling concurrent data access and user sessions.
2. UI/UX Design: Creating a user-friendly interface that was also responsive to different screen sizes took time and multiple iterations. Ensuring intuitive navigation and a smooth user experience was a learning process.
3. Testing and Debugging: Integrating modules like login, registration, and cart handling led to some unexpected bugs, especially when managing user authentication and database synchronization.
4. Time Management: Balancing our workload, especially with the testing phase, was a challenge. Debugging often took longer than expected, which impacted our timeline.

➤ Outcome

The outcome of the project was a functional healthcare application that successfully implemented the core features we aimed to deliver. Users could register, log in, search for doctors, book lab tests, purchase medicines, and read health articles. While the app met our expectations in terms of functionality, there is always room for improvement in areas such as performance optimization and UI consistency.

➤ Reflection and Learnings

This project taught us several valuable lessons:

- **Effective Collaboration:** Working collaboratively and managing version control were essential skills we strengthened. The use of GitHub taught us how to handle merge conflicts and manage different branches effectively.
- **Importance of Testing*** Testing needs to be integrated throughout the development process, not just at the end. Regular testing and debugging sessions helped us catch issues early, and we learned the value of maintaining detailed logs and documentation for testing.
- **User-Centric Design:** This project highlighted the importance of designing with the user in mind. We learned to balance aesthetics with functionality and aim for simplicity in our designs.

➤ Future Improvements

In future projects, we would:

- **Prioritize More Rigorous Testing:** Establish a more structured testing strategy, potentially involving automated testing, to identify bugs early and save time in the later stages.
- **Optimize Database Management:** Experiment with different database solutions to handle more complex data interactions and improve efficiency.

Overall, this project was a rewarding experience that helped us grow as developers, and we are confident that our learnings will lead to even better outcomes in future projects.

12. Appendix:

- **12.1 Full Code Repository:**
<https://github.com/vidhie1/healthcare.git>

Grading Criteria:

- **Design and UI:** 15 marks
- **Functionality:** 20 marks
- **Innovation & Creativity:** 10 marks
- **Technical Complexity:** 15 marks
- **Security and Data Management:** 10 marks
- **Testing and Debugging:** 10 marks
- **Team Collaboration:** 5 marks
- **Documentation:** 5 marks
- **Demo & Viva:** 10 marks